

SERIAL TRANSMISSION SYSTEM

Publication number: JP62075857 (A)

Publication date: 1987-04-07

Inventor(s): NAKAJIMA YASUYUKI

Applicant(s): TOKYO SHIBAURA ELECTRIC CO

Classification:

- international: G06F13/32; G06F13/00; G06F13/28; H04L29/10; G06F13/00; G06F13/20; H04L29/10; (IPC1-7): G06F13/00; G06F13/28

- European:

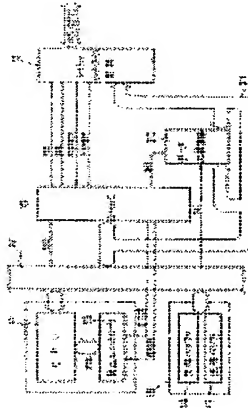
Application number: JP19850216555 19850930

Priority number(s): JP19850216555 19850930

Abstract of JP 62075857 (A)

PURPOSE: To attain the serial transfer of data at high speed and also to improve the processing capacity of a CPU, by using a direct memory access controller for serial input/output control to transfer the information on a memory buffer in blocks.

CONSTITUTION: In a data transfer mode a CPU 11 first sets a message to be transferred to a transmission buffer 17 and also sets the number of transfer data to a direct memory access controller DMAC19. Thus the DMAC19 is started to read the transfer data of the buffer 17 out of a reading circuit 13 regardless of the CPU 11 according to the write signal WE and the transmission ready signal TXRDY sent from a serial interface circuit 13. Then a transmission end signal EXTR is delivered to an interruption controller 15 when the transfer is through with the prescribed number of transfer data.; Thus the CPU 11 knows the end of transmission and sets the next transmission data to the buffer 17. Then these actions are repeated.



⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A)

昭62-75857

⑬ Int.Cl.⁴

G 06 F 13/00
13/28

識別記号

3 5 1
3 1 0

庁内整理番号

7218-5B
7165-5B

⑭ 公開 昭和62年(1987)4月7日

審査請求 有 発明の数 1 (全6頁)

⑮ 発明の名称 直列伝送方式

⑯ 特 願 昭60-216555

⑰ 出 願 昭60(1985)9月30日

⑱ 発 明 者 中 嶋 保 幸 北九州市小倉北区下到尾津1丁目10番1号 株式会社東芝北九州工場内

⑲ 出 願 人 株 式 会 社 東 芝 川崎市幸区堀川町72番地

⑳ 代 理 人 弁理士 鈴 江 武 彦 外2名

明 細 書

1. 発明の名称

直列伝送方式

2. 特許請求の範囲

RS-232C シリアルインターフェイスによりデータの直列転送を行なうものにおいて、CPU と、この CPU にシステムバスを介して接続されるダイレクトメモリアクセスコントローラと、上記 CPU にシステムバスを介して接続されるシリアルインターフェイス回路と、上記 CPU にシステムバスを介して接続されるメモリ部と、上記ダイレクトメモリアクセスコントローラから出力される送信終了信号および受信終了信号に基づいて上記 CPU 11 に割込み信号を出力する割込みコントローラとを具備し、データの受信時には上記ダイレクトメモリアクセスコントローラの制御により受信データを上記メモリ部に記憶し、受信終了後上記割込みコントローラによって CPU にデータの受信を知らせ、CPU が処理中の動作を終了してからメモリ部に記憶したデー

タに基づく処理を行ない、データの送信時には送信データを予め上記メモリ部に記憶し、上記ダイレクトメモリアクセスコントローラの制御により上記 CPU の処理動作とは独立にデータの送信を行なうことを特徴とする直列伝送方式。

3. 発明の詳細な説明

〔発明の技術分野〕

この発明は、EIA (Electronic Industries Association) により規定された RS-232C シリアルインターフェイスによるデータ転送に係わるもので、特にプロセス制御コンピュータ等のデータのシリアル転送時における CPU の処理能力を向上できる直列伝送方式に関する。

〔発明の技術的背景とその問題点〕

一般に、RS-232C でデータのシリアル転送を行なう場合は、例えば第4図に示すような構成で行なわれる。図において、11はCPU、12はシステムバス、13はシリアルインターフェイス回路、14は送信および受信バッファで、上記 CPU 11 によりシステムバス 12 を介して

シリアルインターフェイス回路13が制御される。

上記のような構成において、データの送受は、シリアルインターフェイス回路13のステータスデータをシステムバス12を介してCPU11が監視して行ない、受信の場合はシリアルデータが受信バッファに入ったことを、送信の場合は送信バッファがレディ状態になったことをCPU11が認識した後、データの送受を行なっている。この場合のフローチャートを第5図に示す。第5図において、(a)図は受信時、(b)図は送信時のフローチャートである。

上述したような構成は、主としてシングルタスクで動作する安価なパーソナルコンピュータおよび簡単な制御コンピュータ等に使用されるものであり、CPU11はシリアル通信を行なう場合、シリアルインターフェイス回路13のステータス監視に専念する必要がある。これは例えば、シリアル転送スピード(Baud Rate)が9600BPSとし、1つのデータを10ビット

(スタートビット+データビット(8ビット)+ストップビット)とすれば、約1μsecの間CPU11はシリアルインターフェイス回路13のステータス監視でループすることになり(第5図(a),(b)参照)、データ転送数に比例してCPU11のスループットが低下する欠点がある。

このような欠点を除去でき、CPUの処理能力を向上できるシリアル転送方式として、第6図に示すようなものが提案されている。すなわち、前記第4図においてはCPU11がシステムバス12を介してシリアルインターフェイス回路13のステータス状態を監視していたが、そのかわりに、シリアルインターフェイス回路13から出力される受信レディ信号RXRおよび送信レディ信号TXRを割込みコントローラ15に入力し、割込みコントローラ15から出力される割込み信号PIGによってCPU11が受信レディあるいは送信レディの事象が発生したことを認識できるようにしている。このような構成により、CPU11は各信号(受信レディ信号RXRお

よび送信レディ信号TXR)がアクティブになるまでは他のプログラム(タスク)が実行できることになる。これを第7図(a),(b)に模式化して示す。上記第6図に示す構成は、マルチタスクで動作するパーソナルコンピュータやマルチユーザのソフト開発装置などに使用される。

ところで、上記第6図に示す構成において、マルチタスク動作を行なわせる場合、実際にはリアルタイムマルチタスクOS(RTOS)が使用されるのが一般的であり、各タスクは入出力処理(この場合はシリアル入出力処理)をOSに依存し、OS内で処理する場合が多い。そして、あたかも複数のプログラム(タスク)が同時に動作しているように処理させる事が可能であり、前記第4図の構成に比べCPU11のスループットを向上できる。しかし、OS内での動作を考えた場合、シリアル入出力の1データ毎に発生する受信または送信レディの事象割込み処理時間および各タスク間の切換え時間(モニタのオーバーヘッドタイム)が問題となり、マルチタ

スクのメリットを充分発揮できない場合が生ずる。

[発明の目的]

この発明は上記のような事情に鑑みてなされたもので、その目的とするところは、シリアル転送時におけるシリアル入出力に必要とするCPUの処理時間を最少におさえることができ、且つマルチタスク動作上で生ずるモニタのオーバーヘッド時間をも無視できる程度まで小さくできるすぐれた直列伝送方式を提供することである。

[発明の概要]

すなわち、この発明においては、上記の目的を達成するために、シリアル入出力制御に用いられるダイレクトメモリアクセスコントローラ(DMAC)を使用し、メモリバッファに蓄えられていた情報をブロック転送を行なうことにより高速シリアル転送およびCPUの処理能力の向上を図っている。

〔発明の実施例〕

以下、この発明の一実施例について図面を参照して説明する。第1図におけるシステムバス12には、CPU11、受信バッファ16および送信バッファ17から成るメモリ部18、ダイレクトメモリアクセスコントローラ19が接続される。また、上記ダイレクトメモリアクセスコントローラ19、データ比較回路20およびシリアルインターフェイス回路13は、ローカルデータバス21を介して接続される。上記CPU11からダイレクトメモリアクセスコントローラ19には、上記システムバス12を介してダイレクトメモリアクセスセット信号SDが供給され、このダイレクトメモリアクセスコントローラ19からシリアルインターフェイス回路13には読み込み信号REおよび書き込み信号WEが、また割込みコントローラ15には送信終了信号ETXRおよび受信終了信号ERXRが出力される。上記データ比較回路20には、上記CPU11からシステムバス12を介して比較デ

ータSCが供給され、その比較出力SOがダイレクトメモリアクセスコントローラ19に出力される。そして、上記シリアルインターフェイス回路13からダイレクトメモリアクセスコントローラ19には、受信レディ信号RXRDYおよび送信レディ信号TXRDYが出力され、このシリアルインターフェイス回路13と外部機器との間でRS-232Cにより信号の送受が行なわれる。

次に、上記のような構成において動作を説明する。まず、CPU11が処理を終了してリセットされると、このCPU11からシステムバス12を介してダイレクトメモリアクセスコントローラ19にセット信号SDが出力され、ダイレクトメモリアクセスコントローラ19の初期設定が行なわれるとともに、このダイレクトメモリアクセスコントローラ19の制御によりシリアルインターフェイス回路13の初期設定が行なわれる。この時、シリアル転送フォーマットによりターミネータデータ(転送区切りのデータ)が決定されている場合には、CPU11

の制御により上記比較回路20にターミネータデータが設定される。上述したような初期設定の終了後、CPU11は他のタスクプログラムの実行に入る。そして、他のタスクプログラムの実行中にシリアルデータの受信が発生した場合には、シリアルインターフェイス回路13とダイレクトメモリアクセスコントローラ19との間の受信レディ信号RXRDYと読み込み信号REとに基づき、ダイレクトメモリアクセスコントローラ19がシリアルデータを読み込み、予め設定された受信バッファ16に順次蓄えて行く。この時、CPU11はシリアル転送(この場合は受信モード)に関しての処理は不要であるので、他のタスクプログラムの実行が継続される。そして、上記受信バッファ16へのデータの読み込みが予め設定されたデータ数に達した場合には、ダイレクトメモリアクセスコントローラ19から割込みコントローラ15に受信終了信号ERXRが出力され、この割込みコントローラ15から出力される割込み信号PIGにより

CPU11は受信終了を知ることができる。また、上記データ比較回路20にターミネータデータが設定されている場合には、受信バッファ16に格納されるデータがこのデータ比較回路20により1つのデータ転送毎に比較され、その比較出力SOによりデータがターミネータデータであることが判定されると、ダイレクトメモリアクセスコントローラ19から割込みコントローラ15に受信終了信号ERXRが出力される。そして、上記割込みコントローラ15から出力される割込み信号PIGによりCPU11は受信の終了を知る。

その後、CPU11は必要な時に(例えば現在実行しているタスクプログラムの終了後)受信バッファ16を参照し、転送メッセージを解釈する。

一方、データの送信を行なう場合には、CPU11はまず転送すべき転送メッセージを送信バッファ17にセットするとともに、ダイレクトメモリアクセスコントローラ19に転送データ

数を設定してこのコントローラ19を作動させる。その後、CPU11は他のタスクプログラムの実行に入る。動作が開始されたダイレクトメモリアクセスコントローラ19は、CPU11とは無関係にシリアルインターフェイス回路13からの送信レディ信号TXRDYおよび書き込み信号WEに基づき、順次メモリ部18内の送信バッファ17に格納された転送データを読み出し、シリアルインターフェイス回路13から出力する。データ転送の終了は、予め設定された転送データ数を転送した後、ダイレクトメモリアクセスコントローラ19から割込みコントローラ15に送信終了信号ETXRを出力することによりCPU11に送信の終了が知らされる。この時点で、CPU11は次の送信データを送信バッファ17にセットし、上述した動作を繰り返せば良い。

このような構成によれば、CPU11とは独立して動作するダイレクトメモリアクセスコントローラ19およびメモリ部18を設けたので、

時と送信開始時の各々1度だけであるため、オーバーヘッド時間を短縮できる。上述した差異は、モニタの動作速度及びシリアルデータの転送数により変化するが、一般的には10%以上の処理能力の向上となる。

〔発明の効果〕

以上説明したようにこの発明によれば、シリアル転送時におけるシリアル入出力に必要とするCPUの処理時間を最少におさえることができ、且つマルチタスク動作上で生ずるモニタのオーバーヘッド時間をも無視できる程度まで小さくできるすぐれた直列伝送方式が得られる。

4.図面の簡単な説明

第1図はこの発明の一実施例に係わる直列伝送方式について説明するための図、第2図および第3図はそれぞれ従来およびこの発明の一実施例に係わる直列伝送方式の動作を比較して示す図、第4図ないし第7図はそれぞれ従来の直列伝送方式について説明するための図である。

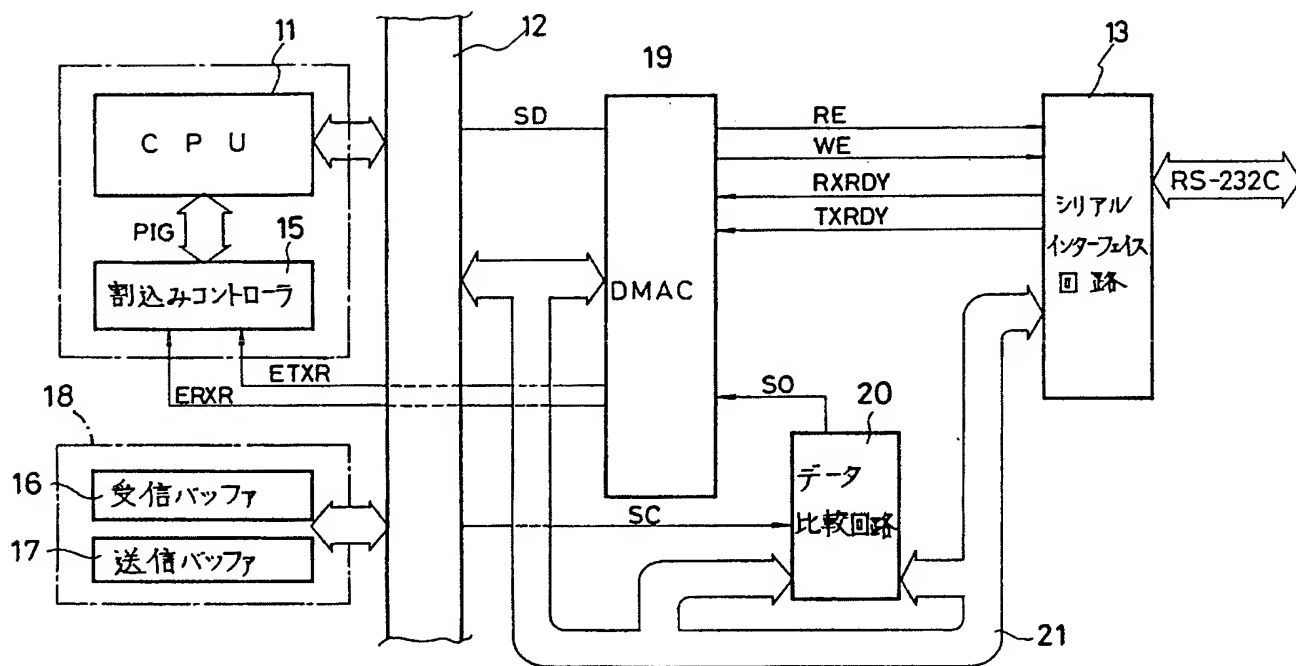
11…CPU、12…システムバス、13…シ

リアルタイム動作中におけるモニタのオーバーヘッド時間を短縮でき、メインCPU11のスループットを向上できる。

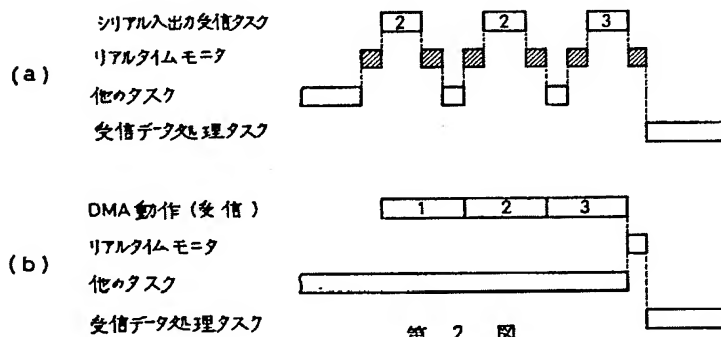
第2図(a)、(b)および第3図(a)、(b)はそれぞれ、前記第1図および前記第6図の構成における各タスクとリアルタイムモニタ間のプログラムの動きを示している。第2図(a)、(b)はデータ受信モード、第3図(a)、(b)はデータ送信モードを示しており、(a)図は前記第6図の構成、(b)図は前記第1図の構成の場合である。これらの図では、3キャラクタの送受信を行なうものと仮定している。図示するように、前記第6図の構成では転送されるキャラクタ数の倍の数だけリアルタイムモニタの制御が必要となる。この時、リアルタイムモニタ内での処理は割込みとタスク切換えの処理であり、これがモニタのオーバーヘッド時間となる。これに対し、前記第1図の構成では、CPU11の動作と並列にダイレクトメモリアクセスコントローラ19がシリアル転送を行ない、モニタの制御に戻る場合は受信終了

リアルタイムインターフェイス回路、15…割込みコントローラ、18…メモリ部、19…ダイレクトメモリアクセスコントローラ、ETXR…送信終了信号、ERXR…受信終了信号、PIG…割込み信号。

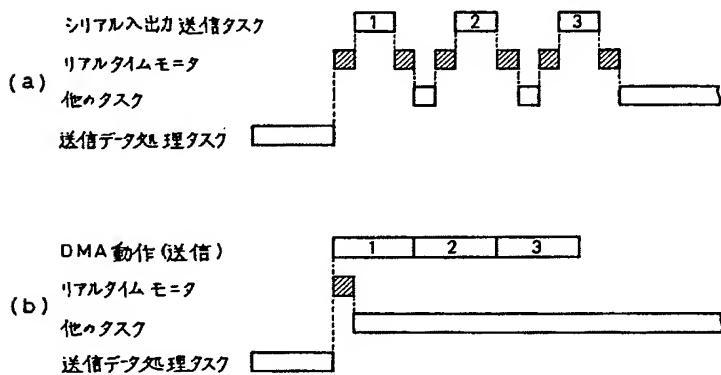
出願人代理人 弁理士 鈴 江 武 彦



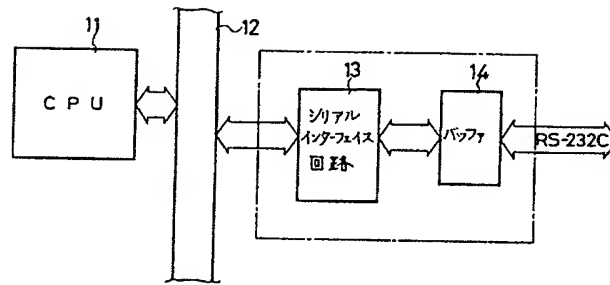
第 1 図



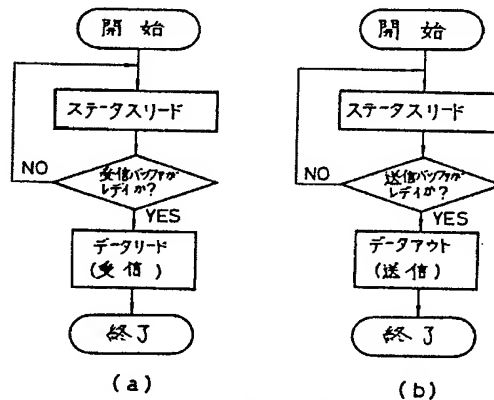
第 2 図



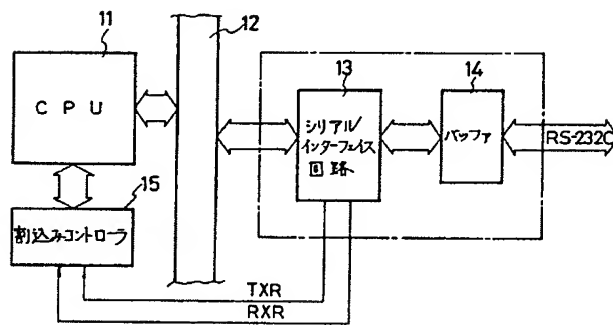
第 3 図



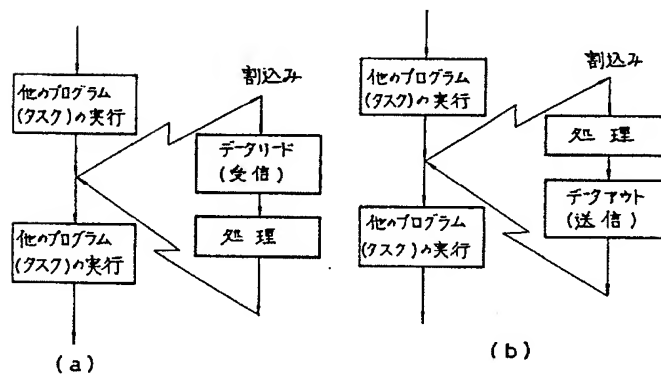
第 4 図



第 5 図



第 6 図



第 7 図